

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété
Intellectuelle
Bureau international



(43) Date de la publication internationale
19 février 2004 (19.02.2004)

PCT

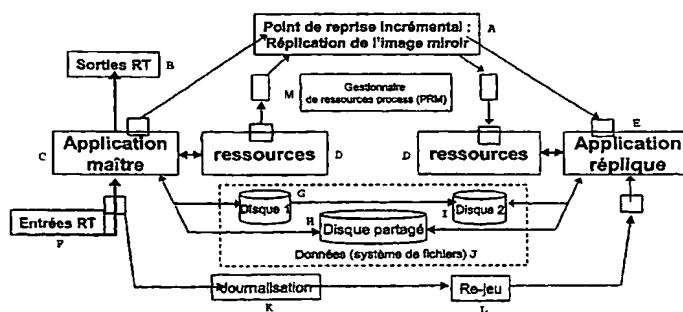
(10) Numéro de publication internationale
WO 2004/015574 A3

- (51) Classification internationale des brevets⁷ : G06F 11/20 (72) Inventeurs; et
(21) Numéro de la demande internationale : PCT/FR2003/002371 (75) Inventeurs/Déposants (pour US seulement) : VERTES, Marc [FR/FR]; 57, rue de Terris, F-31830 Plaisance du Touch (FR). DUFOUR, Laurent [FR/FR]; 3, impasse des Noisetiers, F-31830 Plaisance du Touch (FR). RICHARD, François [FR/FR]; 2, impasse des Garrabiers, F-31120 Lacroix Falgarde (FR). KURZ, Gregory [FR/FR]; 56, route de Tarbes, F-31170 Tournefeuille (FR).
(22) Date de dépôt international : 28 juillet 2003 (28.07.2003)
(25) Langue de dépôt : français
(26) Langue de publication : français
(30) Données relatives à la priorité : 02/09855 2 août 2002 (02.08.2002) FR
(71) Déposant (pour tous les États désignés sauf US) : MEIOSYS [FR/FR]; Centre Industriel d'Innovation de Basso Cambo, 42, avenue du Général de Crouette, F-31100 Toulouse (FR).
(81) États désignés (national) : AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX,

[Suite sur la page suivante]

(54) Title: FUNCTIONAL CONTINUITY BY REPLICATING A SOFTWARE APPLICATION IN A MULTI-COMPUTER ARCHITECTURE

(54) Titre : CONTINUITÉ DE FONCTIONNEMENT PAR REPLICATION D'UN LOGICIEL DANS UNE ARCHITECTURE MULTI-ORDINATEURS



- A INCREMENTAL RESTART POINT: MIRROR IMAGE REPLICATION
B RT OUTPUTS
C MASTER APPLICATION
D RESOURCES
E REPLICATED APPLICATION
F RT INPUTS
G DISK 1
H SHARED DISK
I DISK 2
J DATA (FILE SYSTEM)
K LOGGING
L REPLAY
M PROCESS RESOURCE MANAGER (PRM)

(57) Abstract: Disclosed is a method for replicating a software application in a multi-computer architecture (cluster). Said software application is executed on a first computer of said cluster, which represents a primary or operational node, and is replicated on at least one other computer of the cluster, which represents a secondary node, comprising replication of the resources associated with said software application. The inventive method comprises streamlined updating of the replicated resources by means of a dynamic introspection mechanism supplying the structure of the application that is to be replicated and the dynamic graph of the implemented resources and dependencies.

[Suite sur la page suivante]

WO 2004/015574 A3



MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

— avec revendications modifiées

(84) États désignés (*régional*) : brevet ARIPO (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), brevet eurasién (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), brevet européen (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), brevet OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

(88) Date de publication du rapport de recherche internationale:

2 septembre 2004

Date de publication des revendications modifiées:

16 décembre 2004

Publiée :

— avec rapport de recherche internationale

En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.

(57) Abrégé : Procédé pour répliquer une application logicielle dans une architecture multi-ordinateurs (cluster), cette application logicielle étant préalablement exécutée sur un premier ordinateur du cluster constituant un nœud primaire ou opérationnel et étant destinée à être répliquée sur au moins un autre ordinateur du cluster constituant un nœud secondaire, comprenant une réplification des ressources associées à cette application logicielle. Ce procédé comprend une mise à jour au fil de l'eau des ressources répliquées par un mécanisme d'introspection dynamique prévu pour fournir la structure de l'application à répliquer, ainsi que le graphe dynamique des ressources et dépendances mises en œuvre.

REVENDICATIONS MODIFIEES

[reçues par le Bureau international le 21 septembre 2004 (21.09.04);
revendications originales 1 et 5 remplacées par la revendication modifiée 1;
revendications originales 2-4 inchangées ; revendications 6-20 inchangées mais renumérotées
5-19 (4 pages)]

1. Procédé pour répliquer une application logicielle dans
une architecture multi-ordinateurs (cluster), ladite
5 application logicielle étant préalablement exécutée sur un
premier ordinateur dudit cluster constituant un nœud
primaire et étant destinée à être répliquée sur au moins un
autre ordinateur dudit cluster constituant un nœud
secondaire, comprenant une réplication des ressources
10 associées à ladite application logicielle, caractérisé en ce
que les ressources répliquées incluent :

- la mémoire virtuelle de chaque processus concerné ainsi
que sa pile d'appel,
- des ressources systèmes (inter process communication,
15 connexion réseau, etc.), et
- des données écrites sur disques;

et en ce qu'il comprend en outre une mise à jour au fil de
l'eau desdites ressources répliquées par un mécanisme
d'introspection dynamique prévu pour fournir la structure de
20 l'application à répliquer, et un graphe dynamique des
ressources et dépendances mises en œuvre.

2. Procédé de réplication selon la revendication 1,
caractérisé en ce qu'il comprend en outre une création et
25 une maintenance d'un arbre de dépendance, qui fournit à
chaque instant des informations sur les ressources qu'il est
nécessaire de répliquer.

3. Procédé de réplication selon l'une des revendications 1
30 ou 2, caractérisé en ce qu'il comprend en outre un mécanisme
de génération de point de reprise (« checkpointing »), par
lequel les ressources à répliquer sont répliquées sur un ou
plusieurs nœuds secondaires.

4. Procédé de répllication selon la revendication 3, caractérisé en ce qu'il comprend trois étapes :

- capture des ressources sur le nœud primaire,
- transfert par le réseau vers un ou plusieurs nœuds secondaires, et
- restauration sur le ou les nœuds secondaires.

5. Procédé de répllication selon l'une quelconque des revendications précédentes et la revendication 3, caractérisé en ce qu'il comprend en outre un mécanisme d'optimisation du mécanisme de génération de point de reprise.

6. Procédé de répllication selon la revendication 5, caractérisé en ce que le mécanisme de « checkpointing » est incrémental.

7. Procédé de répllication selon l'une des revendications 5 ou 6, caractérisé en ce que le mécanisme de « checkpointing » est discriminant.

8. Procédé de répllication selon l'une des revendications 5 à 7, caractérisé en ce que le mécanisme de « checkpointing » inclut au moins l'une des fonctions suivantes :

- une barrière de synchronisation de processus (PSB),
- une gestion de ressources (RM),
- une gestion de ressources système (SRM), et
- une gestion de ressources de processus (PRM).

9. Procédé de répllication selon l'une quelconque des revendications précédentes, caractérisé en ce qu'il comprend en outre un mécanisme de répllication de fichiers de données applicatives entre un nœud opérationnel (OP) sur lequel l'application est exécutée et un nœud dit de stand-by (SB).

10. Procédé pour réaliser une continuité de fonctionnement d'une application logicielle dans une architecture multi-ordinateurs (cluster), cette application étant exécutée à un instant donné sur l'un des ordinateurs du cluster, appelé nœud primaire ou opérationnel, les autres ordinateurs dudit cluster étant appelés nœuds secondaires, ce procédé mettant en œuvre le procédé de réplication selon l'une quelconque des revendications précédentes,
- 10 caractérisé en ce qu'il comprend les étapes suivantes :
- réplication de l'application sur au moins des nœuds secondaires, de façon à réaliser au moins un clone de ladite application,
 - mise à jour au fil de l'eau dudit ou desdits clones,
- 15 et
- en cas de détection d'une défaillance ou d'un événement affectant ledit nœud opérationnel, basculement de service vers l'un au moins desdits clones.
- 20 11. Procédé de continuité de fonctionnement selon la revendication 10, caractérisé en ce que la réplication de l'application est de nature holistique.
12. Procédé de continuité de fonctionnement selon l'une 25 revendication 10 ou 11, caractérisé en ce qu'il comprend en outre une mise à jour des clones de l'application.
13. Procédé de continuité de fonctionnement selon l'une des revendications 10 à 12 , caractérisé en ce qu'il comprend en 30 outre une supervision de l'état de ressources nécessairement au fonctionnement de l'application.
14. Procédé de continuité de fonctionnement selon l'une des revendications 10 à 13, caractérisé en ce qu'il comprend en

outre, à la suite d'une détection d'une défaillance ou d'un événement affectant le nœud opérationnel, une étape pour élire, parmi des clones installés sur des nœuds secondaires, un clone pour être substitué à l'application initiale, le
5 nœud sur lequel ledit clone élu est installé devenant le nouveau nœud opérationnel.

15 15. Procédé de continuité de fonctionnement selon l'une des revendications 10 à 14, caractérisé en ce qu'il comprend en
10 outre un enregistrement sur chaque clone de messages reçus par le nœud primaire ou opérationnel, ces messages étant réinjectés dans le clone élu nouvel opérationnel en cas de basculement.

15 16. Système multi-ordinateurs prévu pour exécuter sur au moins desdits ordinateurs au moins une application
logicielle, implémentant le procédé pour réaliser une continuité de fonctionnement selon l'une quelconque des revendications 11 à 15.

20 17. Application du procédé de réplication selon l'une quelconque des revendications 1 à 9, pour une optimisation automatique de ressources informatiques par partage de charge par répartition dynamique de processus.

25 18. Application du procédé de réplication selon l'une quelconque des revendications 1 à 9, pour une maintenance non interruptive par relocation à la demande de processus au travers d'un réseau de ressources informatiques.

30 19. Application du procédé de réplication selon l'une quelconque des revendications 1 à 9, pour une préservation de contexte applicatif dans des applications mobiles.